Ibáñez, R., Ammar, A., Cueto, E., Huerta, A., Duval, J.-L., and Chinesta, F.
"Multi Scale Proper Generalized Decomposition based on the Partition of Unity"
to appear in the *International Journal for Numerical Methods in Engineering*

Multi Scale Proper Generalized Decomposition based on the Partition of Unity

Rubén Ibáñez[1,4], Amine Ammar[2], Elías Cueto[3], Antonio Huerta[4],
Jean-Louis Duval[5], and Francisco Chinesta[6]

[1]*ICI - High Performance Computing Institute, Ecole Centrale de Nantes, 1 rue de la Noe, F-44300 Nantes, France*
[2]*LAMPA, ENSAM Angers, 2 Boulevard du Ronceray, BP 93525, 49035 Angers, France*
[3]*Aragon Institute of Engineering Research, Universidad de Zaragoza, Edificio Betancourt. Maria de Luna, s.n., 50018 Zaragoza, Spain*
[4]*Laboratori de Calcul Numeric, Universitat Politecnica de Catalunya, BarcelonaTech, Calle Jordi Girona 1, 08034 Barcelona, Spain*
[5]*ESI Group, 3bis rue Saarinen, 94528 Rungis, France*
[6]*ESI Group Chair @ PIMM - Procédés et Ingénierie en Mécanique et Matériaux, ENSAM ParisTech, 151 boulevard de l'Hôpital, Paris, France*

SUMMARY

Solutions of partial differential equations could exhibit a multi-scale behavior. Standard discretization techniques are constraint to mesh up to the finest scale in order to predict accurately the response of the system. The proposed methodology projects the solution of the PDE into a set of parent unidimensional spaces which are solved following the standard PGD rationale. The proposed methodology can be seen as an alternative to circumvent prohibitive meshes arising from the necessity of capturing fine scale features.

Received ...

KEY WORDS: Partition of unity, Proper Generalized Decomposition, Time multi-scale.

1. INTRODUCTION

Many problems in computational mechanics present a multi-scale behavior where the micro-scale effects influence the macro-scale ones and viceversa. Hence, the treatment of the different scales of the problem becomes very important to reach an accurate solution. Plenty of effort has been dedicated to deal with small scale effects either refining the mesh or introducing models that account for subgrid effects. Turbulence modeling can be seen as a typical example where different alternatives can be implemented to account for subgrid effects, namely, Reynolds Averaged Navier-Stokes (RANS), Large-Eddy Simulation (LES) or Direct Navier-Stokes (DNS) [9].

Such a multi-scale behavior in the spatial domain also appears when dealing with metamaterials whose structure is defined at the micro level. Several techniques have been applied throughout the history like the so-called homogenization [17], defining the micro-scale constitutive model in terms of a representative volume that satisfies the Hill-Mandel principle [10]. This methodology has been proven to be very effective in order to circumvent the prohibitive computational cost of methods like FEM^2 [10]. However, there is a strong hypothesis behind such homogenization technique, requiring a clear separation of scales between macro and micro effects.

Another way to handle these multi-scale effects is by using the variational multi-scale framework introduced by [24]. Here the effect of the micro-scale into the macro-scale variables is introduced in a consistent way developing different weak forms associated to the macro and micro scales. This approach has been effectively applied in many problems such as Navier-Stokes equation [25] among others.

Moreover, the multi-scale problem will appear when dealing with temporal evolutions that are composed by many harmonics at very different frequencies. Following standard time marching approaches, a suitable time step that captures the evolution of the finest scale has to be imposed, deriving into a prohibitive simulation cost. Many numerical algorithms have been proposed to avoid a time marching approach capturing up to the finest time scale.

Neron et al. [22] proposed a numerical algorithm based on the LATIN-PGD able to handle temporal and spatial multi-scale behaviors appearing in solid mechanics problems. However, selecting the temporal macro basis just like the interfacial degrees of freedom coupling different macro domains could be problem dependent. The scalability of the methodology is restored via an appropriate correction of the temporal basis based on the residual. Fritzen et al. [11, 21] also partitioned the time domain into subintervals, where a common reduced basis is applied for all subintervals. Afterwards, extra interface restrictions have to be imposed to ensure continuity of the primal variable and her time derivative. Hence, variables at both extremes of each macro interval are set to zero, plus an offset calculated from the previous macro interval. Several viscoelastic problems are solved for various cyclic loading with varying frequency. Maday et al. [15, 23] developed the so-called *Pararreal* algorithm that also attempts to solve a temporal evolution iterating between macro and micro domain partitions so that the initial conditions for each micro interval are given by the macro resolution of the problem. The algorithm is also highly parallelizable, making it very efficient for either long time simulations or temporal problems involving many spatial degrees of freedom. Ammar et al. in [1] proposed an efficient technique based on the separation of variables, where the time domain was partitioned into a number of subdomains following the standard PGD rationale [4, 5, 19]. However, special attention had to be paid at the interfaces between discrete subdomains, where continuity was imposed by means of Lagrange multipliers. Badias et al. in [3] developed an algorithm where the time domain was partitioned into different PGDs, but again the interface conditions between different subdomains could become a cumbersome task. Another numerical strategy based on combining different PGDs under the partition of unity (PU) rationale [2, 16, 26] was developed in [14]. The idea of using macro shape functions that respect the partition of unity allowed the method to combine different PGDs in a consistent manner. Moreover, the overlap between PGDs automatically solves the problem of interface compatibilities related to the not overlapped methodology. However, there may be situations in which the macro domain partition requires many PGDs to capture properly the behavior, reducing the scalability of the method.

The methodology proposed in this work aims to treat multi-scale problems by means of the separation of variables, where no need to impose a clear separation between scales is required. Indeed, the methodology is based on two main pillars: a generalization of the idea of time domain partition, together with the overlapped PU-based multi-PGD framework. By doing that, the proposed methodology naturally inherits continuity between macro subdomains, clear extrapolation to higher dimensional spaces and scalability of the method even if the macro-domain partition is fine enough.

2. BASICS OF THE METHOD

This section explains the main characteristics of the multi-scale PGD based on the partition of unity. The formulation is introduced for a one-dimensional variable first. Then, the formulation is extended to 2D problems and finally to time-multiscale problems.

Let us assume that a given function, $u(x)$, is the approximation of the solution of a given partial differential equation. In terms of standard approximation basis, such as finite elements, it could be expressed as

$$u(x) = \sum_{i=1}^{N} u_i \, N_i(x), \tag{1}$$

where N stands for the number of degrees of freedom used in the approximation of $u(x)$ or, in other words, the number of nodes in the mesh. Needless to say, $N_i(x)$ represent the standard finite element shape functions and u_i the unknown nodal values.

However, if the solution presents a multi-scale features, the mesh has to capture the details of the solution at the finest scale, thus deriving into a prohibitive simulation cost ($N \gg 1$). A possible way to circumvent this issue, within the PU paradigm, is to introduce a dependent variable that captures the solution details associated to the finest scale,

$$u(x) = \sum_{i=1}^{N} u_i \, N_i(x) \sum_{j=1}^{J} g_j \, G_j\big(\tau(x - x_i)\big), \tag{2}$$

where x_i is the centroid of the shape function $N_i(x)$, $\tau(x - x_i)$ is a dependent variable that presents an offset based on x_i, $G_j(\tau)$ is the j-th micro-scale shape function and g_j its associated micro-scale degree of freedom. Indeed, the micro-scale effects occurring in the compact support of the macro shape function $N_i(x)$ are going to be mapped into a micro-scale parent space, τ.

Remark 1
It is worth noting that in Eq. (2) the coefficients of macro degrees of freedom (u_i) and micro degrees of freedom (defining the local micro enrichment, g_j) are both unknown, in opposition to usual applications of PU where the enrichment functions are known and given a priori.

Figure 1 shows the shape functions associated to both the macro (top) and the micro scale (bottom). Notice how a two-scale approach presents two meshes related to micro and macro scales, respectively. Indeed, each macro partition of the domain contributes to the parent micro scale and the parent micro space affects equally the macro partition.

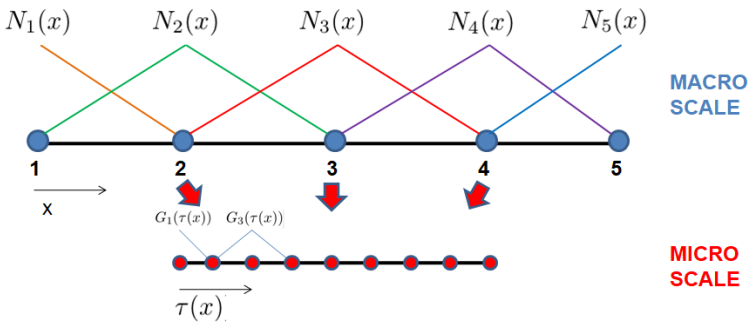

Figure 1. Shape functions of a multi-scale approach. Top, macro shape functions. Bottom, micro shape functions.

Moreover, it is worth mentioning the possibility of rewriting our solution function as $u(x, \tau)$, making it suitable for a PGD-like algorithm [6, 12]. In this case, we proceed with a separation of variables as shown in Figure 2. Therefore, the initial one dimensional problem is transformed into a two dimensional problem. It is important to highlight that the macro mesh does not have to capture the solution of the finest scale, since it is handled by the micro-scale mesh.

Thus, following a standard PGD rationale [7,8], the solution is sought in a Greedy manner through of a finite sum of M modal enrichments,

$$\begin{aligned} u(x,\tau) \approx u^m(x,\tau) &= \sum_{m=1}^{M} \sum_{i=1}^{N} u_i^m N_i(x) \sum_{j=1}^{J} g_j^m \, G_j\big(\tau(x - x_i)\big) \\ &= \sum_{m=1}^{M} \sum_{i=1}^{N} u_i^m N_i(x) \, [\mathbf{g}^m]^T \mathbf{G}\big(\tau(x - x_i)\big), \end{aligned} \tag{3}$$

where the super-index m indicates the m-th PGD mode.

The discretization of a variational formulation by means of FEM, for instance, also needs an approximating space for the admissible variation of the field. In this particular case, when computing

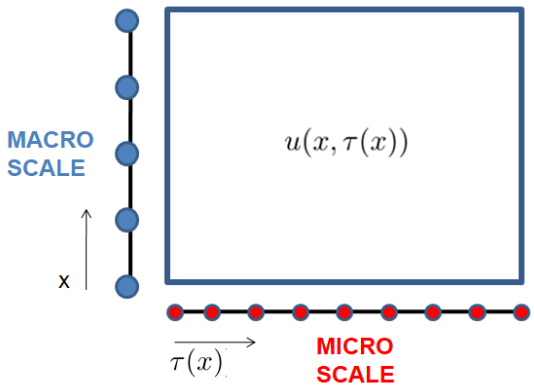

Figure 2. A separation of dependent variables, $u(x,\tau(x))$.

the M-th mode, we used a standard Galerkin projection

$$w^*(x,\tau) = \sum_{i=1}^{N} u_i^* N_i(x)\,[\mathbf{g}^M]^T \mathbf{G}\big(\tau(x-x_i)\big) + \sum_{i=1}^{N} u_i^M N_i(x)\,[\mathbf{g}^*]^T \mathbf{G}\big(\tau(x-x_i)\big). \tag{4}$$

Nevertheless, SUPG or GLS stabilization can be also implemented in this formulation when dealing with highly convective (parabolic, in general) problems, see also [13].

Even if any linearization technique is possible, an alternated direction scheme is chosen in this work for solving the resulting nonlinear problem arising from the PGD rationale. Hence, when solving micro-scale modes, macro-scale admissible variations are set to zero (i.e., $u_i^* = 0$) and, conversely, when solving macro-scale modes, $\mathbf{g}^* = \mathbf{0}$.

Special attention needs to be paid when dealing with the derivatives of the approximation space, viz.

$$\begin{aligned} \frac{Du^m(x,\tau)}{Dx} &= \frac{\partial u^m(x,\tau)}{\partial x} + \frac{\partial u^m(x,\tau)}{\partial \tau}\frac{\partial \tau}{\partial x} \\ &= \sum_{m=1}^{M}\sum_{i=1}^{N}\Big(u_i^m \frac{\partial N_i(x)}{\partial x}[\mathbf{g}^m]^T\mathbf{G}(\tau) + u_i^m N_i(x)\,[\mathbf{g}^m]^T \frac{\partial \mathbf{G}(\tau)}{\partial \tau}\frac{\partial \tau}{\partial x}\Big) \end{aligned} \tag{5}$$

shows the application of the chain rule in order to account for the dependance of $\tau(x-x_i)$. For the sake of simplicity, the dependance of τ with respect to $(x-x_i)$ coordinate will be omitted if there is no risk of confusion.

The same rationale can be easily applied to higher dimensional spaces. Imagine that a 2D case (i.e., $\mathbf{x} = (x,y)$) is approximated by means of a 2D macro mesh and the details of the micro scale are given by one-dimensional modes acting along each spatial direction. Hence, the approximation space will read

$$u(\mathbf{x},\tau_x,\tau_y) \approx u^m(\mathbf{x},\tau_x,\tau_y) = \sum_{m=1}^{M}\sum_{i=1}^{N} u_i^m N_i(\mathbf{x})\,[\mathbf{g}^m]^T\mathbf{G}(\tau_x)\,[\mathbf{h}^m]^T\mathbf{H}(\tau_y), \tag{6}$$

where $\mathbf{g}^m$ and $\mathbf{h}^m$ are the degrees of freedom of the m-th PGD mode acting on x and y subscales, respectively. Indeed, u_i adopts the same role than in the 1D case, since it will be responsible of weighting the subgrid behavior throughout the domain.

The separated representation by means of partition of unity-based PGD in higher dimensions is depicted in Figure 3. As it can be noticed, an initial 2D problem is transformed into a 3D problem, where one dimension takes into account the macro variations of the solution, whereas the other two dimensions take into account the subgrid scales associated to the x and y directions, respectively. It

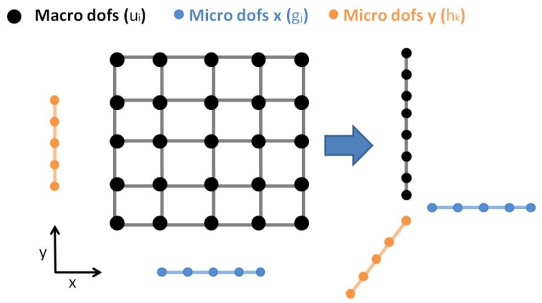

Figure 3. A separation of dependent variables in a 2D space, $u(\mathbf{x}, \tau_x, \tau_y)$.

is very important to notice that subgrid scales are going to propagate throughout the entire domain, being the macro mesh the responsible of controlling the propagation of the subgrid behavior.

It is also important to highlight that the proposed approach reduces the amount of degrees of freedom required to handle subgrid scales in comparison with standard multigrid methods [20]. Imagine that a 2D domain is partitioned using a coarse regular mesh composed of $D_x \times D_y$ linear elements. Let us assume that each element is refined even more to handle subgrid scales using a mesh of $d_x \times d_y$ linear elements. The typical size of the entire system will be $D_x \times D_y \times d_x \times d_y$. Nevertheless, the proposed methodology is able to handle subgrid scales at a computer cost proportional to $M((D_x \times D_y) + d_x + d_y)$. Indeed, if the number of modes required to represent the solution does not grow too much (i.e., $M = \mathcal{O}(10)$), computing subgrid scales becomes affordable.

3. NUMERICAL EXAMPLES

In this section, several numerical examples based on the multi-scale PGD formulation are tested. The first part shows the convergence of the method for three different one-dimensional cases. The second part shows several examples of the proposed methodology for different two-dimensional cases.

3.1. One-dimensional numerical examples

In this first section we analyze three different toy problems in one dimension. These include approximation, diffusion and convection problems.

3.1.1. Multiscale approximation of a given function This first example concerns the approximation of a function that shows multiscale features. Since we deal with an approximation case, there is no governing partial differential equation in this problem. In other words, a given function $f(x)$ is to be approximated by means of a multi-scale approximation $u(x, \tau)$. The weighted residual form associated to this problem reads

$$\int_\Omega w^*(x,\tau)u(x,\tau)dx = \int_\Omega w^*(x,\tau)f(x)dx \quad \forall\, x \in \Omega = [0,1]. \tag{7}$$

In this case the function to approximate is defined as

$$f(x) = \frac{x\cos(8\pi x)}{8\pi}. \tag{8}$$

Figure 4 shows the reconstructed solution (red line) versus the reference solution (blue line) when the macro domain is partitioned using 5 degrees of freedom (black dots), i.e., $N = 5$. The micro domain is partitioned using 80 linear finite elements, i.e., $J = 81$. As it can be noticed, the

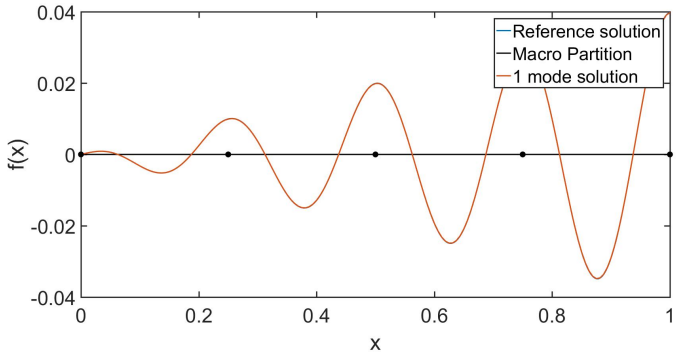

Figure 4. Approximation problem. The reconstructed solution with one single mode is represented by the red line. The reference solution is the blue line. In this case, the macro-domain partition employed 5 degrees of freedom, represented as black dots.

macro partition coincides exactly with the period of the signal, which is precisely the reason why only 1 mode ($M = 1$) is required to accurately represent the solution. Indeed, the micro scale mode is giving us the cosine signal whereas the macro scale is taking care of the linear growth.

However, the exact period of our signal is not known a priori in most cases. Therefore, convergence of the method when the macro partition does not coincide with the signal period is checked as well. Figure 5 shows the reconstructed solution (red line) against the reference one (blue line) when the macro domain partition does not coincide with the signal period. The top figure, involves a reconstructed solution with one mode, whereas the bottom one involves five modes. It is important to notice that the one-mode solution captures the main trend of the signal. However, there are some regions where the signal is not properly captured. Indeed, the solution involving five modes reproduces the reference signal very accurately.

Figure 6 shows the relative error in logarithmic scale between the reconstructed solution and the reference solution for different number of PGD modes. As it can be noticed, the multiscale PGD converges monotonically.

3.1.2. Diffusion Case The second test problem is a pure diffusion equation. The weak form associated to this problem is

$$\int_\Omega \frac{Dw^*(x,\tau)}{Dx}\frac{Du(x,\tau)}{Dx}dx = \int_\Omega w^*(x,\tau(x))f(x)dx, \quad \forall\, x \in \Omega = [0,10]. \tag{9}$$

The chain rule is required, as in Eq. (5), due to the dependence of τ with respect to x. The source term and the two boundary conditions ensuring that the problem is well posed are

$$f(x) = \cos(2\pi x) + \frac{x}{100},\ u(0) = 0,\ \text{and}\ u(10) = 0.$$

Under these conditions, the analytical solution of the problem reads

$$u(x) = \frac{\cos(2\pi x) - 1}{4\pi^2} + \frac{x}{6} - \frac{x^3}{600}.$$

Figure 7 compares the reconstructed solution using the first PGD mode against the reference solution. The macro domain has been partitioned using 8 degrees of freedom. As it can be noticed, the first mode already captures the macro behavior, however, extra modes are required to identify the oscillatory behavior of the micro scale. Indeed, the reconstructed solution with 3 modes already captures the oscillations of the micro scale.

Figure 8 shows the convergence the reconstructed solution with respect to the number of PGD modes. Again, results show good convergence properties towards the reference, exact solution.

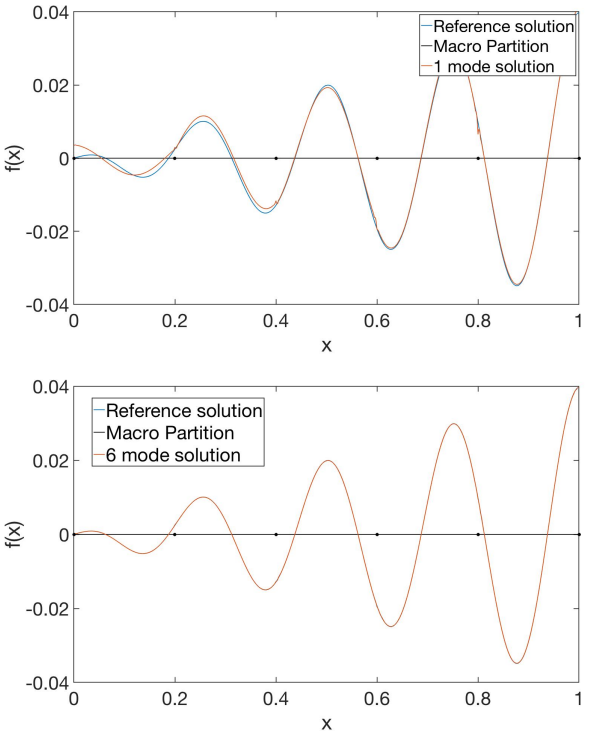

Figure 5. Approximation problem. Top: solution with one single mode. Bottom: 6 modes. The reconstructed solution appears as a red line. The reference solution is represented by a blue line. Macro-domain partition with 6 degrees of freedom, appearing as black dots.

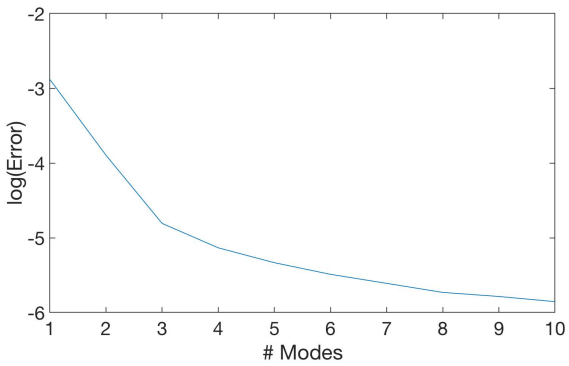

Figure 6. Relative error of the reconstructed solution with respect to the reference solution as a function of the number of PGD modes for the approximation problem.

3.1.3. Convection Case The third example is a convection problem, whose weak form reads

$$\int_\Omega w^*(x,\tau(x))\frac{Du(x,\tau(x))}{Dx}dx = \int_\Omega w^*(x,\tau(x))f(x)dx \quad \forall\, x \in \Omega = [0,10]. \tag{10}$$

It is well-known that special attention has to be paid when dealing with convection-dominated equations to ensure convergence. A standard streamline-upwind Petrov-Galerkin (SUPG) is implemented in this particular case [9]. Therefore, the test function is defined as

$$w^*(x,\tau(x)) = u^*(x,\tau(x)) + \beta\frac{Du^*(x,\tau(x))}{Dx}, \tag{11}$$

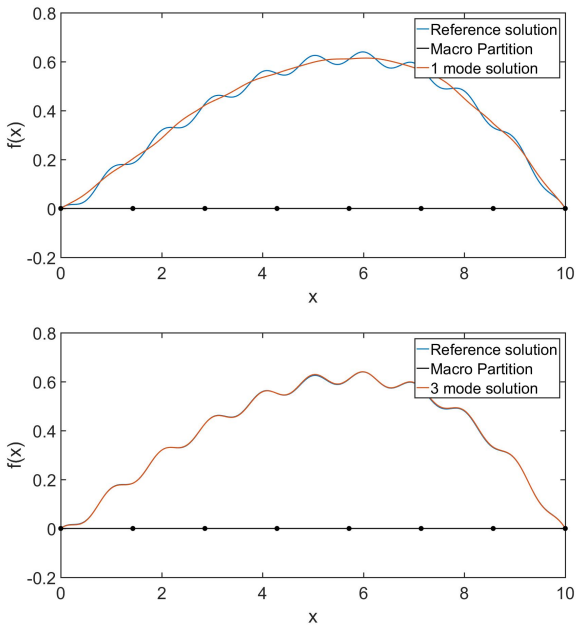

Figure 7. Diffusion problem. Top, 1 mode. Botoom, 3 modes. Reconstructed solution, red line. Reference solution, blue line. Macro domain partition with 8 degrees of freedom, black dots.

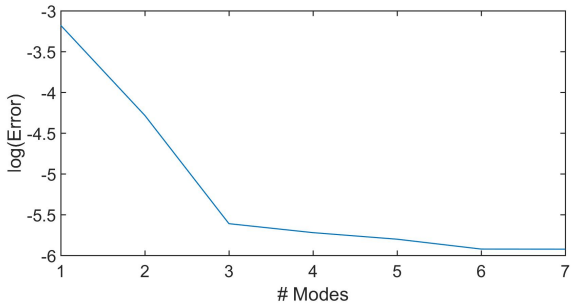

Figure 8. Relative error of the reconstructed solution with respect to the reference solution as a number of PGD modes for the diffusion case.

where β is a numerical coefficient controlling the stabilization of the numerical scheme. It is important to notice that the numerical scheme remains consistent since the test function affects both sides of Eq. (10).

The source term of the problem is

$$f(x) = x\sin(12\pi x) + \frac{x}{30}, \tag{12}$$

while the initial condition 8left boundary) is taken as

$$u(0) = 0. \tag{13}$$

In that case, the analytical solution associated to this initial value problem is

$$u(x) = \frac{\sin(12\pi x)}{144\pi^2} - \frac{x\cos(12\pi x)}{12\pi} + \frac{x^2}{60}. \tag{14}$$

Figure 9 shows the reconstructed solution involving one (top) and seven (bottom) modes versus the reference one. The macro domain has been partitioned using 9 degrees of freedom. As it can be

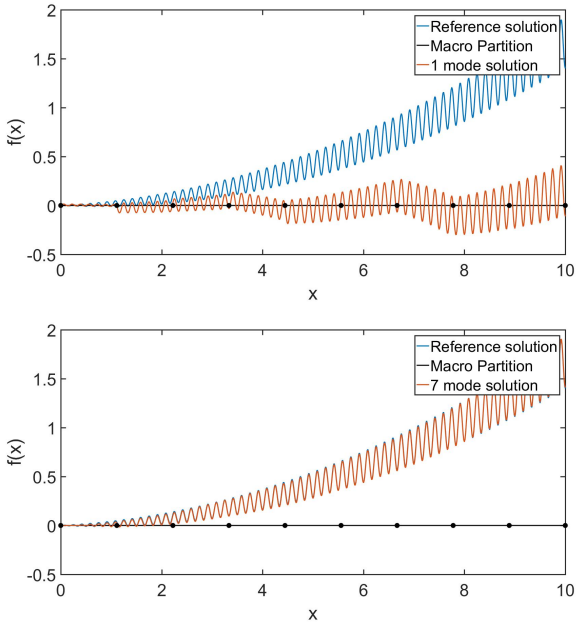

Figure 9. Convection case. Top, 1 mode. Bottom, 7 modes. The reconstructed solution is represented as a red line, while the reference solution appears in blue line. Macro domain partition with 10 degrees of freedom, represented as black dots.

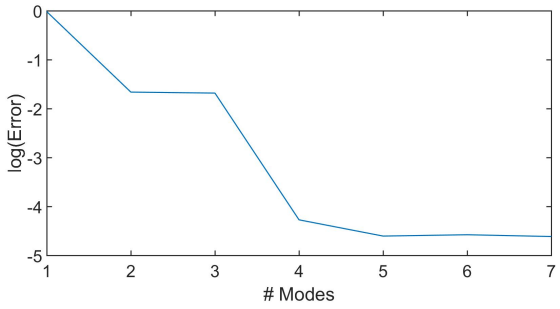

Figure 10. Relative error of the reconstructed solution with respect to the reference solution as a number of PGD modes for the convection case.

noticed, the reconstructed solution is far from the reference one, when only one mode is involved. However, the PGD algorithm converges towards the reference solution when more modes are added into the approximation.

In turn, Figure 10 shows the convergence of the multi-scale PGD algorithm for the 1D convection case. The method converges well, reaching a relative error of 10^{-4} when using 7 modes. It is important to notice that the error seems to stagnate after 5 PGD modes. Of course, by refining the macro scale the error decreases.

3.2. Two dimensional numerical examples

This section extends the analysis of the developed methodology to three different two-dimensional cases: approximation, diffusion and convection-reaction-diffusion problems.

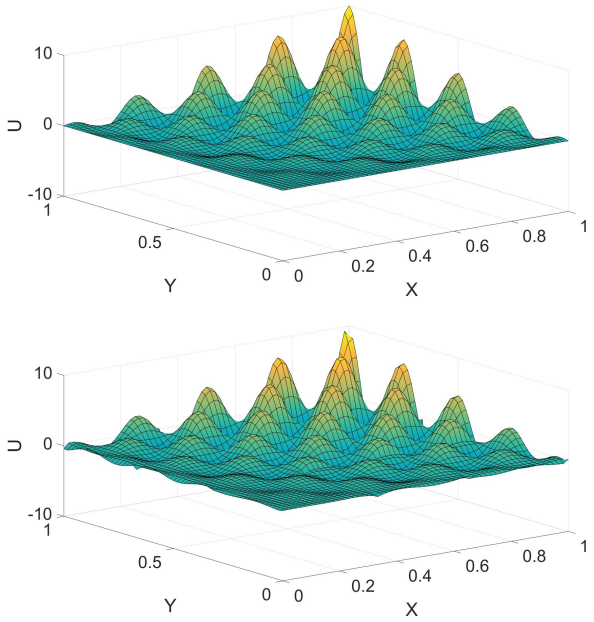

Figure 11. 2D Approximation case. Top: reference solution. Bottom: reconstructed solution. Macro domain partitioned with $5 \times 5 = 25$ degrees of freedom.

3.3. 2D approximation problem

Again, we consider the multiscale approximation of a given function $f(\mathbf{x})$ by a sought function $u(\mathbf{x})$. The weighted residual (Bubnov-Galerkin) form of this problem is

$$\int_\Omega w^*(\mathbf{x}, \tau_x, \tau_y) u(\mathbf{x}, \tau_x, \tau_y) d\mathbf{x} = \int_\Omega w^*(\mathbf{x}, \tau_x, \tau_y) f(\mathbf{x}) d\mathbf{x} \quad \forall\, \mathbf{x} \in \Omega = [0,1]^2 \tag{15}$$

In this case we consider a function

$$f(\mathbf{x}) = x \cos(8\pi x) y \cos(8\pi y). \tag{16}$$

Note that no boundary conditions need to be imposed for this problem.

Figure 11 shows the analytical solution (right) versus the reconstructed solution using the multiscale PGD algorithm with 9 modes. The macro domain has been partitioned using a mesh of 25 degrees of freedom, since each spatial direction has been partitioned using 5 nodes. As it can be clearly seen, the reconstructed solution captures the main features of the reference solution. The small differences are due to the fact that the partition of the macro domain does not coincide with the period of the signal. Therefore, extra modes will be required to alleviate these small oscillations.

Figure 12 shows the convergence plot associated to the 2D approximation case as a function of the modal enrichments of the solution. Note the monotone convergence of the solution.

3.4. 2D diffusion case

The weak form associated to this diffusion problem is

$$\int_\Omega \frac{Dw^*(\mathbf{x}, \tau_x, \tau_y)}{D\mathbf{x}} \cdot \frac{Du(\mathbf{x}, \tau_x, \tau_y)}{D\mathbf{x}} d\mathbf{x} = \int_\Omega w^*(\mathbf{x}, \tau_x, \tau_y) f(\mathbf{x}) d\mathbf{x} \quad \forall\, \mathbf{x} \in \Omega = [0,1]^2. \tag{17}$$

with boundary conditions

$$u(0, y) = u(1, y) = u(x, 0) = u(x, 1) = 0. \tag{18}$$

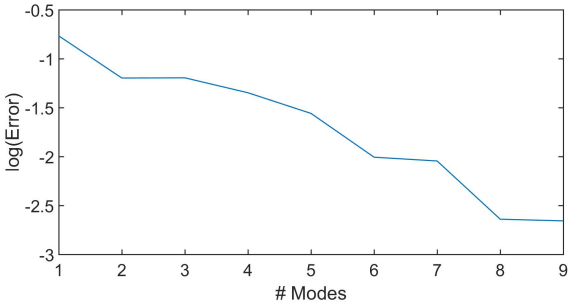

Figure 12. Relative error of the reconstructed solution with respect to the reference solution as a number of PGD modes for the 2D approximation case.

The chain rule is required, as shown in Eq. (5), due to the dependence of τ_x and τ_y with respect to x and y, respectively. The source term f considered in this case was

$$f(\mathbf{x}) = 2w^2xy\sin(wx)\sin(wy) - 2wx\sin(wx)\cos(wy) - 2wy\cos(wx)\sin(wy). \tag{19}$$

In this particular case, the angular velocity is set to $w = 20\pi$. In that situation, the analytical solution is given by

$$u(x, y) = x\sin(wx)y\sin(wy). \tag{20}$$

It is important to highlight the treatment of the boundary conditions in order to solve the diffusion problem. Vanishing Dirichlet boundary conditions need to be imposed on the boundary, so that all macro degrees of freedom placed at the boundary are fixed to zero. To enforce non-vanishing Dirichlet boundary conditions, the first modes will satisfy the Dirichlet boundary conditions whereas the subsequent modes will be computed with vanishing boundary conditions, following the standard rationale of the PGD [12].

Figure 13 shows the reference solution for the diffusion case (top) versus the reconstructed solution using 5 modes of the PGD. The macro domain has been partitioned using a mesh of 4 nodes per direction but since the nodes at the boundary are set to zero, the final degrees of freedom are $2 \times 2 = 4$. Note that there is no perceivable difference between the reference solution and the reconstructed one.

3.5. 2D convection-diffusion problem

The weak form associated to this diffusion problem is

$$\int_\Omega w^*(\mathbf{x}, \tau_x, \tau_y)\frac{Du(\mathbf{x}, \tau_x, \tau_y)}{Dy}d\mathbf{x} + \int_\Omega \frac{Dw^*(\mathbf{x}, \tau_x, \tau_y)}{Dx}\frac{Du(\mathbf{x}, \tau_x, \tau_y)}{Dx}d\mathbf{x} = \int_\Omega w^*(\mathbf{x}, \tau_x, \tau_y)f(\mathbf{x})d\mathbf{x}, \ \forall\, \mathbf{x} \in \Omega = [0, 1]^2 \tag{21}$$

Note that along the y direction there is a pure convection phenomenon, whereas the x direction presents pure diffusion. Hence, the nature of the PDE forces to impose boundary conditions at both ends of the x interval, whereas only initial boundary conditions must be imposed in the y direction. The source term reads in this case

$$f(\mathbf{x}) = (16w^2y^2 + 2y)\sin(4wx) + (64w^2 - w)\sin(8wx)\sin(wy) \tag{22}$$

while the set of boundary conditions ensuring the well-posedness of the problem reads

$$u(0, y) = u(1, y) = u(x, 0) = 0. \tag{23}$$

In this case the angular velocity is set to $w = \pi$.

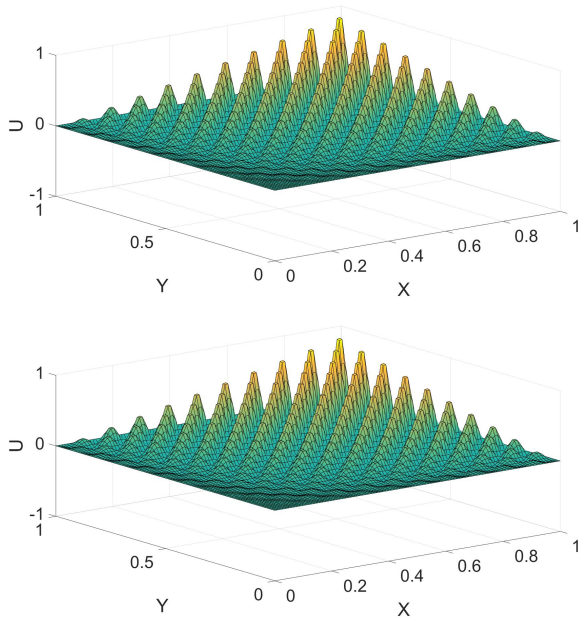

Figure 13. 2D Diffusion case. Top: reference solution. Bottom: reconstructed solution with 5 modes. Macro domain partitioned with $2 \times 2 = 4$ degrees of freedom.

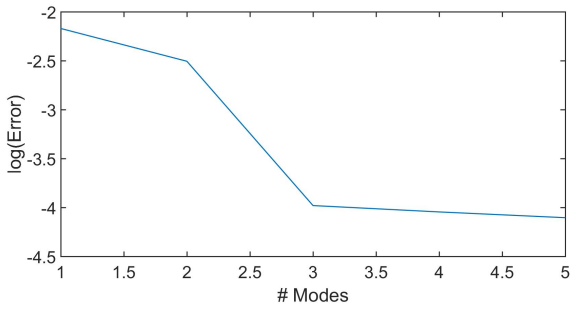

Figure 14. Relative error of the reconstructed solution with respect to the reference solution as a number of PGD modes for the 2D diffusion case.

Under this set of conditions, the partial differential equation (21) admits a unique solution given by

$$u(\mathbf{x}) = \sin(4wx)y^2 + \sin(8wx)\sin(wy). \tag{24}$$

Figure (15) shows the reference solution (top) versus the reconstructed solution involving 5 modes (bottom). The macro domain has been partitioned using $3 \times 4 = 12$ degrees of freedom. As it can be noticed, the reconstructed solution already captures the main features of the reference solution. It is important to notice how the reconstructed solution is able to reproduce the set of homogeneous boundary conditions imposed in the 3 sides of the squared domain.

Figure 16 shows the relative error versus the number of modes involved in the reconstructed solution. As it can be seen, the error decreases with respect to the number of modes, showing good convergence properties even for convection diffusion problems.

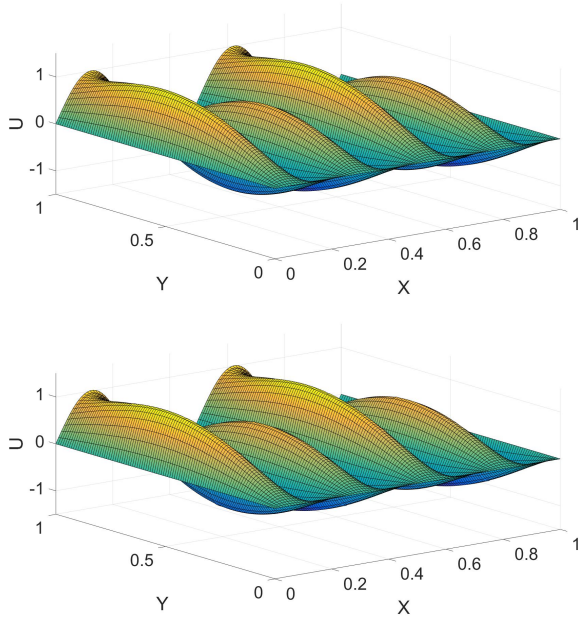

Figure 15. 2D Convection-Diffusion case. Top: reference solution. Bottom: reconstructed solution with 5 modes. Macro domain partitioned with $3 \times 4 = 12$ degrees of freedom.

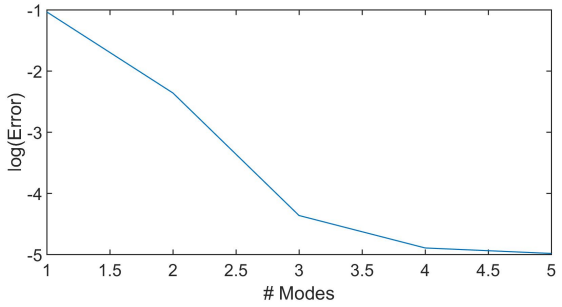

Figure 16. Relative error of the reconstructed solution with respect to the reference solution as a number of PGD modes for the 2D convection diffusion case.

3.6. Time Multi-Scale applied to a discrete system of equations

The main aim of this example is to show the capabilities of the algorithm to predict the response of a transient system of equations arising from a standard FEM discretization in space. The strong form of the problem reads

$$\frac{\partial u(x,t)}{\partial t} - \frac{\partial^2 u(x,t)}{\partial x^2} = f(x,t). \tag{25}$$

Indeed, the problem analyzed in this section makes reference to a 1D transient thermal problem subjected to the following boundary conditions and source term

$$\begin{aligned} f(x,t) &= (4\pi^2(sin(t) + sin(50t)) + cos(t) + 50cos(50t))sin(2\pi x) \\ u(x,0) &= 0 \\ u(0,t) &= u(1,t) = 0 \end{aligned} \tag{26}$$

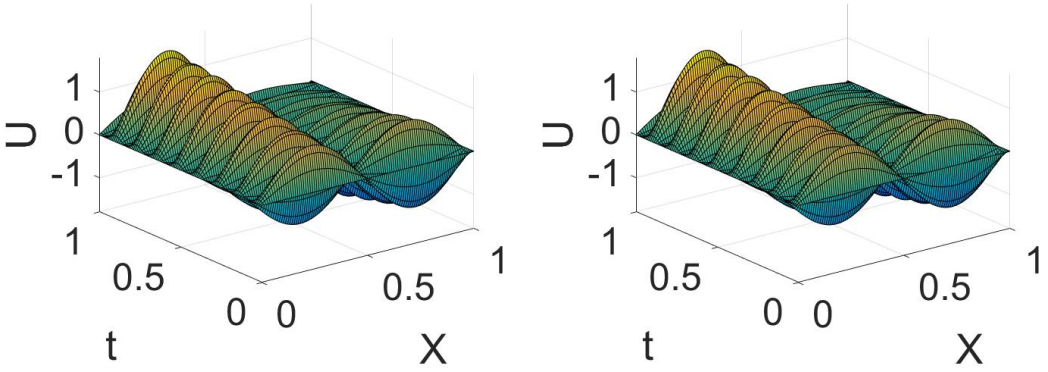

Figure 17. Transient thermal problem. Reference solution, left. Reconstructed solution with 1 mode, right. Temporal domain partitioned with 4 degrees of freedom.

This partial differential equation under this set of boundary conditions admits an analytical solution

$$u(x,t) = (sin(t) + sin(50t))sin(2\pi x), \tag{27}$$

which presents a two scale behavior in the time domain.

If the spatial 1D domain is discretized using 80 linear elements equally spaced, it gives the discrete system of equations

$$\mathbf{M}\dot{\mathbf{u}}(t) + \mathbf{K}\mathbf{u}(t) = \mathbf{f}(t) \in t = [0,T], \tag{28}$$

where multi-scale is applied only in time.

Figure 17 shows the reference solution (left) and the reconstructed solution (right) using 1 mode for the discretized transient thermal problem. As it can be noticed, the obtained solution is in good agreement even when involving only one mode. This fact is not surprising since the analytical solution is composed only by one separated mode. Indeed, we found that only one mode produces a relative error with respect to the analytical solution of $1e-5$.

4. CONCLUSIONS

A novel algorithm able to solve multi-scale problems in an efficient way is proposed in this work. The algorithm combines a macro partition of the domain which is enriched within the PU rationale. The resulting multi-scale multi-dimensional model is solved using the PGD separated representation. The convergence of the algorithm is proven to be very effective for 1D and 2D problems, involving algebraic, diffusion and convection phenomena.

The methodology renders good results even when the macro domain partition does not involve a pure periodical signal in the micro scale. The extension of the algorithm to higher dimensional spaces is our current line of research just like the extension of the methodology to non-rectangular domains.

A. ELEMENTAL OPERATORS

The main aim of the appendix is to give details about the construction of the operators required to make the multi-scale PGD approximation for the different two-dimensional PDEs analyzed in the paper. It is worth to mention that the potential of the PGD algorithm is achieved when all expressions appearing in the formulation are separable. Therefore, we will assume that the macro

shape functions appearing in a 2D problem are separable,

$$N_i(x,y) = N_i^x(x)N_i^y(y). \tag{29}$$

Note that this is the case, for instance, of linear rectangular elements. Nevertheless, the methodology could be applied to more complex elements, involving more terms in the separation of the shape function.

A.1. Diffusive Operator

In this subsection a diffusion of a scalar field $u(x,y)$ along the x direction is developed. The derivation for the diffusion along the y direction follows a similar rationale. For the sake of simplicity but without loosing generality, we will also assume that a new mode in the macro direction is desired. Therefore, all variations related to the micro x and y direction are set to zero.

$$\int_{\Omega_e} \frac{Du^*}{Dx}\frac{Du}{Dx}dxdy = \sum_{i=1}^{4}\sum_{j=1}^{4}\sum_{k=1}^{M} u_i^* \int_{\Omega_e} \left(\frac{\partial N_i}{\partial x}X_M Y_M + N_i \frac{\partial X_M}{\partial x}Y_M\right)\left(\frac{\partial N_j}{\partial x}X_k Y_k + N_j \frac{\partial X_k}{\partial x}Y_k\right)u_j dxdy$$

$$\begin{aligned}\int_{\Omega_e} \frac{Du^*}{Dx}\frac{Du}{Dx}dxdy = &\sum_{i=1}^{4}\sum_{j=1}^{4}\sum_{k=1}^{M} u_i^* u_j \int_x X_M \frac{\partial N_i^x}{\partial x}\frac{\partial N_j^x}{\partial x}X_k dx \int_y Y_M N_i^y N_j^y Y_k dy \\ &+ \sum_{i=1}^{4}\sum_{j=1}^{4}\sum_{k=1}^{M} u_i^* u_j \int_x \frac{\partial X_M}{\partial x} N_i^x \frac{\partial N_j^x}{\partial x}X_k dx \int_y Y_M N_i^y N_j^y Y_k dy \\ &+ \sum_{i=1}^{4}\sum_{j=1}^{4}\sum_{k=1}^{M} u_i^* u_j \int_x \frac{\partial X_M}{\partial x} N_i^x N_j^x \frac{\partial X_k}{\partial x} dx \int_y Y_M N_i^y N_j^y Y_k dy \\ &+ \sum_{i=1}^{4}\sum_{j=1}^{4}\sum_{k=1}^{M} u_i^* u_j \int_x X_M \frac{\partial N_i^x}{\partial x} N_j^x \frac{\partial X_k}{\partial x} dx \int_y Y_M N_i^y N_j^y Y_k dy \end{aligned} \tag{30}$$

It can be highlighted that four different contributions are appearing due to the application of the chain rule. Furthermore, all terms appearing in the last part of Eq. (30) already present a separated format.

A.2. Convective Operator

In this subsection a pure convection of a scalar field $u(x,y)$ along the x direction is derived. It is important to notice that this case is a particular case of a general convection given by the velocity field $\mathbf{v}$, where a separated representation of the velocity field would be required. The derivation for the convection along the y direction is given as an exercise for the reader. For the sake of simplicity but without loosing generality, we will also assume that a new mode in the macro direction is desired, being the micro scale directions known.

$$\int_{\Omega_e} u^* \frac{\partial u}{\partial x} dxdy = \sum_{i=1}^{4}\sum_{j=1}^{4}\sum_{k=1}^{M} \int_{\Omega_e} u_i^* N_i X_M Y_M \left(\frac{\partial N_j}{\partial x} X_k Y_k + N_j \frac{\partial X_k}{\partial x} Y_k \right) u_j dxdy$$
$$= \sum_{i=1}^{4}\sum_{j=1}^{4}\sum_{k=1}^{M} u_i^* u_j \int_x X_M N_i^x \frac{\partial N_j^x}{\partial x} X_k dx \int_y Y_M N_i^y N_j^y Y_k dy$$
$$+ \sum_{i=1}^{4}\sum_{j=1}^{4}\sum_{k=1}^{M} u_i^* u_j \int_x X_M N_i^x N_j^x \frac{\partial X_k}{\partial x} dx \int_y Y_M N_i^y N_j^y Y_k dy. \quad (31)$$

Once again all integrals appearing in the last part of Eq. (31) are written already in a separated way, improving the efficiency of the algorithm.

REFERENCES

1. A. Ammar, F. Chinesta, E. Cueto, M. Doblare. *Proper generalized decomposition of time-multiscale models*. International Journal for Numerical Methods in Engineering, vol. 90, pp. 569-596, 2012.
2. Babuška, Ivo, and Jens M. Melenk. The partition of unity method. International journal for numerical methods in engineering 40.4 (1997): 727-758.
3. A. Badias, D. Gonzalez, I. Alfaro, F. Chinesta and E. Cueto. *Local Proper Generalized Decomposition*. Numerical Methods in Engineering, vol. 112, 2017.
4. F. Chinesta, G. Chaidron, A. Poitou *On the Solution of the Fokker-Planck Equations in Steady Recirculating Flows Involving Short Fiber Suspensions*. Journal of Non-Newtonian Fluid Mechanics, vol. 113, pp. 97-125, 2003.
5. F. Chinesta, A. Leygue, F. Bordeu, J.V. Aguado, E. Cueto, D. Gonzalez, I. Alfaro, A. Ammar and A. Huerta. *PGD-Based Computational Vademecum for Efficient Design, Optimization and Control.*. Archives of Computational Methods in Engineering, vol. 20, pp. 31-59, 2013.
6. Chinesta, Francisco, Pierre Ladeveze, and Elias Cueto. *A short review on model order reduction based on proper generalized decomposition.* Archives of Computational Methods in Engineering 18, no. 4, 395, 2011.
7. Chinesta, Francisco, and Elias Cueto. *PGD-based modeling of materials, structures and processes*. Heidelberg: Springer, 2014.
8. Cueto, Elias, David Gonzalez, and Iciar Alfaro. *Proper generalized decompositions: an introduction to computer implementation with Matlab*. Springer, 2016.
9. Donea, Jean, and Antonio Huerta. *Finite element methods for flow problems*. John Wiley & Sons, 2003.
10. Feyel F. Multiscale FE^2 elastoviscoplastic analysis of composite structures. Computational Materials Science 1999; 16(1-4):344-354.
11. Fritzen, Felix, and Matthias Leuschner. Reduced basis hybrid computational homogenization based on a mixed incremental formulation. Computer Methods in Applied Mechanics and Engineering 260 (2013): 143-154.
12. Gonzalez, David, Amine Ammar, Francisco Chinesta, and Elias Cueto. *Recent advances on the use of separated representations.* International Journal for Numerical Methods in Engineering 81, no. 5, pp. 637-659, 2011.
13. Gonzalez, David, Elias Cueto, Francisco Chinesta, P. Diez, and Antonio Huerta. *SUPG-based stabilization of Proper Generalized Decompositions for high-dimensional advection-diffusion equations.* Int. J. Numer. Methods Eng 94, no. 13, pp. 1216-1232, 2013.
14. R. Ibáñez, E. Abisset-Chavanne, F. Chinesta, A. Huerta, E. Cueto. *A local, multiple Proper Generalized Decomposition based on the Partition of Unity*. International Journal of Numerical Methods in Engineering, submitted, 2018.
15. Maday, Yvon, and Gabriel Turinici. A parareal in time procedure for the control of partial differential equations. Comptes Rendus Mathematique 335.4 (2002): 387-392.
16. Melenk, Jens M., and Ivo Babuška. The partition of unity finite element method: basic theory and applications. Computer methods in applied mechanics and engineering 139.1-4 (1996): 289-314.
17. J.C. Michel, H. Moulinec, P. Suquet. *Effective properties of composite materials with periodic microstructure: a computational approach*. Computer methods in applied mechanics and engineering, vol. 172, pp. 109-143, 1999.
18. Nron, David, and Pierre Ladeveze. Proper generalized decomposition for multiscale and multiphysics problems. Archives of Computational Methods in Engineering 17.4 (2010): 351-372.
19. S. Niroomandi, I. Alfaro, D. Gonzalez, E. Cueto and F. Chinesta. *Model order reduction in hyperelasticity: a Proper Generalized Decomposition approach*. International Journal for Numerical Methods in Engineering, vol. 00, pp. 1-28, 2012.
20. Trottenberg, Ulrich, Cornelius W. Oosterlee, and Anton Schuller. Multigrid. Elsevier, 2000.
21. F. Fritzen, M. Reza. *Space-time model order reduction for nonlinear viscoelastic systems subjected to long-term loading*. Meccanica. 10.1007/s11012-017-0734-x, 2017.
22. JC. Passieux, P. Ladevèze, D. Nèron. *A scalable time-space multiscale domain decomposition method: adaptive time scale separation*. Computational Mechanics. https://doi.org/10.1007/s00466-010-0504-2, 2010.
23. Y. Maday, M. Riahi, J. Salomon. *Parareal in Time Intermediate Targets Methods for Optimal Control Problems*. Control and optimization with PDE constraints. Based on the international workshop on control and optimization of PDEs, Mariatrost, Austria, 2011.

24. T. Hughes, G. Feijóo, L. Mazzei, J.B. Quincy. *The variational multiscale method: a paradigm for computational mechanics*. Computer Methods in Applied Mechanics and Engineering, vol. 166, pp. 3-24, 1998.
25. R. Codina, S. Badia, J. Baiges, J. Principe. *Variational Multiscale Methods in Computational Fluid Dynamics*. https://doi.org/10.1002/9781119176817.ecm2117, 2017.
26. I. Babuska, J.M. Melenk *The Partition of Unity Method*. International Journal for Numerical Methods in Engineering, vol. 40, pp. 727-758, 1997.